Day1:Introduction:

Definition of Dev OPs: makeover of process, tools, methodologies, starts with business analysts, developers, testers to the guys that manage the analytics.

* Optimizing every process
* Culture – way of doing things in an organization
* Automation(reliable and repeatable ) at all phases of the software development cycle not testing only

How to test better?

* Think from the perspective of the user

DeVOps Activities starts of as the following

* Plan
* Code **focuses on development**
* Build:compiling individual pieces of code developed by individuals to make sure it works as a single piece of code
* Test
* Release
* Deploy **focuses on Operations**
* Operate
* Monitor: monitoring the behavior of the usage pattern of the software and it is the food thought of the next release cycle

The difference that DevOps brings from the normal software development lifecycle is that for every activity that occurs in the process, you have to choose which method or process is best to use for each of the activities that occur in DevOps, i.e. plan, Code, Build

Day 2 10/04/2017

* DevOps – software development method that stresses communication, collaboration and integration between software developers and information technology professionals
* User Interface-representations , how will the application will look
* User Experience-Usability- number of clicks , number of fields
* Pagination – displaying of number of pages per page
* I18N-Internationalization-google
* L10N – Localization – google

Growing Pains

* How do you develop provision for virtual machines: i.e. uses to user’s experience the same experience as when the application is used by one person or 15 million
* Aggregate logs :if 15 millions people using an application how does the organization find out if one user experiences an issue

Operations Complexity

**DEVOPS Principles**

1. **Infrastructure Automation(IA)-**provisioning of Hardware, processor , RAM,AWS-Amazon Web Services – the whole use 70% cloud services-organized sector of services available to an individual, also known as Cloud

* Controls network and hardware
* LAMDA-Processing unit for virtualization
* Examples of public web services, amazon web services, windows azure , rack space cloud, hp j

1. **Configuration Management :**is a set of process which will install your operating system and software automatically

* Operating System
* Software

i.e standard bank having to update the operating system for their employees of over 40 000, instead of going to do it on an individual machine a code is written to update the software automatically.

Examples of tools: Chef, Puppet, Ansible, Salt Stack, Pallet,BCFG2

1. **Deployment Automation**

deploys application into server , talks with software management and deploys

Tools used for Deployment:

* Jenkins
* Fabric
* Capistrano

1. **Log Management**

Reading the log files ,aggregating all messages.

Types of messages/log

* Information message
* Error message
* Severe message

Tools used for Log Management:

* Splunk
* Sumologic
* LogStash

1. **Performance Management**

Types of tools used

* App Dynamics
* Boundary
* CloudWeaver

**Understanding the business need for DevOps**

-

11/04/2017

DEVOPS Principles

Jenkins is a tool for continuous integration and occurs in configuration management

Continuous build and test continuously occur and integration between and build test occur

There are 4 kinds of environment

Shift left- Operational Concerns-more of an introduced development

**SIW**

1. Development environment
2. Test environment
3. Staging-Replica of production, used for UAT and realistic data is used
4. Production – continuous monitoring production events

**CI – Continuous Integration**

Consists of the following

Facilitates continuous integration and Jenkins is used for this

1. Continuous Development

2 .Continuous Testing

3. Continuous Delivery

4 .Continuous Management

Cycle of Continuous Integration

1. Development – Laptop – Eclipse-
2. Commit - Githup
3. Build
4. Test Jenkins
5. Development

DevOps Cycle

Customer

Business owner

Development test

Operations/Production

GITHUB used for SCM

All code from the repository is downloaded to server

Fog-

Repository is divide into branches, and then branches and then master file

SCM-source code management

Build- Virtual studio and eclipse

Maven is used for compilation and dependencies , is a building tool, and this tool produces the below files:

a)Jar-JAVA Archive

b)War- Web Archive

Building your code and compiling it is called build artifact

JRE-is not used in development it is for deploying server-It is a runtime environment

JDK-used in Deployment –it is a

Application server to deploy the application, the prerequisite for this is an application server, example of a tool to deploy is called TOMCAT.

Sanity testing-check first few test case

GITHUB steps

-Created account

-Created a repository

-created a file by multiple ways

1. Online editor

2. Drag and drop

3. Select file or upload

-downloaded the repository

-forked the repository from the other user(bringing other code into another user account)

-Added users as collaborators to a repository

-Clone , create a replica into local laptop

- then installed the Client(Github for windows)

-In the tool did the following:

-Sign in

- went +sign and cloned a repository to bring the code into the laptop

-created a file and checked the diff(difference) not committing to something one is not suppose to do

- and then commiting only to the local

-the publish

-or push

-merge

- or sync

-diff(

1. clone

2. sync

3. diff

4. commit

5. sync

6. push

**Introducing Continuous Integration in your organization**

**Phases of CI**

Phase 1

Phase 2

Phase 3**-**Nightly builds

Phase 4-Enter the metrics

Phase 5 –Getting more serious about testing

Phase 6- Automated UAT and more Automated Deployment

Phase 7 –Continuous Deployment

**Jenkins SetUp**

* Can be used for Java and non-java protects
* Continuous integration tool
* Open source tool(source code is available to the whole world on the internet, people should be allowed to contribute to it,it’s a myth that its free, )
* Has multiple plug ins

**Steps in Jenkins**

1. **Manage Jenkins**
2. Global tool Configuration tool
3. Add JDK
4. Go to windows Explore and then copy file path on JAVA and paste into Java\_Home Field
5. Deselect Install automatically
6. Enter “myMaven”on Maven
7. Copy Apache maven 3.3.9 and paste it on “Maven\_Home”
8. Click Save

**Creating Job**

1. New Item

2. Type “game of life”

3. Click on” Free style object”

4. Click Ok

5. Check “Discard old build”

6. Enter Max number of builds as “3”

7. Click on “Apply”

8. 7. Add build step

9. Choose “invoke top level maven targets”

10. maven version and select”my maven”

11. goals choose “install”

12. click on “save”

13.”click on build now”

14. Click on “console out”

15. go to “Users”and select your users-jenkins-Workspace-gameoflife

16. copy all files

17. Open another windows explorer and go to the” game of life master”

18. Paste it into the game if life master

19. Go back to the Jenkins tool

20. Click on “Build now”

21. Click on “Console output”

22. Go to C-Windows Explorer-Windows-system32-Config-System profile

23. Open new window-c-window-users-your id-.M32-Copy all .m32 directory

24. Paste this into System profile

25. repeat step 20-21

26.